Using the ez-Cryptosystem for Data Transmission in Virtual Private Networks (Vpn)

Amirgaliyev Yedilkhan[1],
[1]*Suleyman Demirel University*
Almaty, Kazakhstan
amir_ed@mail.ru

Amanzholova Saule, Kalizhanova Aliya,
Zamanova Saule, Kozbakova Ainur[2]
[2]*Kazakh National Research Technical University after K.I. Satpaev*
Almaty, Kazakhstan
shokataeva@gmail.com, kalizhanova_aliya@mail.ru,
saule_zamanova@mail.ru, ainur79@mail.ru

***Abstract* —The aim of the article is to research the process of information security in transmission between virtual subnets which are realized on data encryption algorithms of EZ-cryptosystem and secret key that protects the information from interception. In fact, the data to be intersegmental transfer coded output from one network, and decoded at the other input network, wherein the data encryption algorithm allows secure distribution between their endpoints. All data manipulations are transparent to the user working on the network.**

Keywords—VPN, EZ-cryptosystem, encryption, decryption.

I. Introduction

Today local area network (LAN) administrators are facing with the problems of information protection more often. It is especially important for networks where you are working with the information that should be protected additionally. It can be networks of governmental/public institutions, companies that produce certain kind of goods, other organizations that can be damaged economically if they have any disclosures. The problem becomes more serious when networks should be available to the Internet access or to mobile users and staff of remote offices also if LAN users have an opportunity to be connected to the Internet.

II. EZ-cryptosystem

Let Σ be the finite set, which is called an alphabet. Elements from Σ are called letters, finite sequence of elements from Σ is called a word. The length of a word w is a number of letters, which are in w. Set Σ^* of words is called the space of source codes. If X is some set, then $|X|$ is the number of elements of a set X. Number $|\Sigma|$ is the length of alphabet Σ. If Σ is an alphabet, then the original text which is an element of Σ is called alphabetic original text (apt). Similarly, cipher text of any element from Σ is called alphabetic cipher text (act). Then the length of original text coincides with the number of alphabetic original text, contained in it and the length of cipher text coincides with the number of alphabetic cipher text, contained in it.

It is constructed an infinite family of Euclidean cryptosystems satisfying the condition:

(A) The encryption algorithm allows encrypting the source code of any length so that after encoding with a fixed encryption key all encrypted alphabetic texts are different, i.e. the cipher text contains no repetitions.

In condition (A) the key expressions are "any length" and "fixed key", since there exist other well-known cryptosystems without repetitions, called polyalphabetic cryptosystems. From classical cryptosystems such concerns Vigenere cipher, of the modern - CBC modes and CTR modes. But if the key is fixed, then the length of the text without repetition is limited. Consider, for example CBC. Let $\{0,1\}^m$ - set of sequences of length m, consisting of zeros and ones. If the key is $(a_1, a_2, ..., a_m) \in \{0,1\}^m$, it is easy to see that if the length of the text exceeds $l \cdot 2^m$, where l the length of the alphabet, then there are repetitions.

Assume that there is a cipher text with the large length. Then one of the methods for finding alpha source is the statistical analysis, which lies at the base of the frequency distribution of alphabet letters. Obviously, these methods are not applicable to cryptosystems satisfying condition (A) even theoretically.

The main theorem

Algorithm of encryption and decryption comes from:

Theorem 1. Let k be an arbitrary integer number. Then for any $p \in Z$, such that 1<|p|<|k| and for any $s \in Z$, such that

$$\gcd(s,k)=1,\ \gcd(s,p)=1 \tag{1}$$

There exist integers a_p, c, satisfying the following conditions:

1) $|a_p| < |k|$.

2) $c = a_p s$, $c \equiv p \pmod{k}$.

3) $|k| < |c|$.

4) If $p_1 \neq p_2$ and $|p_1| < |k|, |p_2| < |k|$, then $c_1 \neq c_2$, and inverse, if $c_1 \neq c_2$, then $p_1 \neq p_2$.

Proof: Let k be an arbitrary integer number, p - integer number, satisfying the condition $|p| < |k|$. As $\gcd(s,k) = 1$, that of the Chinese remainder theorem tells about existing of integer number b_p , such that

$$b_p \equiv 0 (\bmod s), \quad b_p \equiv p (\bmod k) \qquad (2)$$

and $b_p = pus$, where the number u may be taken from equality $us + vk = 1$. Let a_p is the remainder after dividing the $d_p = pu$ by k . It is known that b_p , satisfying the conditions (2) uniquely defined, but from condition $\gcd(s,k) = 1$ it is clear, that any two numbers of the form $d'_p = pu', d''_p = pu''$ have the same remainders from dividing by k , i.e. a_p uniquely determined. Indeed, suppose that b'_p, b''_p satisfy the conditions (2). Then we have

$$b'_p - b''_p = d'_p s - d''_p s \equiv 0 (\bmod k) .$$

Therefore $d'_p - d''_p \equiv 0 (\bmod k)$, as well as

$$|a'_p|, |a''_p| < |k|$$

$$a'_p - a''_p \equiv d'_p - d''_p \equiv 0 \pmod{k}$$

and $a'_p = a''_p$.

It is easy to see that

$$b_p = d_p s \equiv a_p s (\bmod k) \equiv p (\bmod k) \qquad (3)$$

If $c = a_p s$, then from (3) follows that c satisfies the condition (2) from the theorem. Inequality $|k| < |c|$ is right, since otherwise $c = a_p s \equiv p \pmod{k}$, $|p| < |k|, |c| < |k|$

and $c = a_p s = p$, but it is impossible, as $\gcd(s, p) = 1$ and condition (3) from the theorem is hold. From (2) follows that if $|p_1|, |p_2| < |k|$ and $p_1 \neq p_2$, then $c_1 \neq c_2$ and if $c_1 \neq c_2$, then $p_1 \neq p_2$, hence condition (4) of the theorem holds.

III. Encryption and decryption algorithms

(C_1) Algorithm of encryption: let k be an integer number, p an integer number such that $|k| > |p|$ and s any integer number, satisfying conditions (1). Then from the theorem 1 we get c . So, we have: k - secret key, p - plain text, c - cipher text. From the proof of theorem 1 follows that the algorithm of encryption consists of next steps:

Step 1: find u , using the equality $us + vp = 1$. It is well known that u can be found generalized by the Euclidean algorithm.

Step 2: find a_p in the form of remainder from dividing pu by k .

Step 3: $c = a_p s$.

(D_1) Algorithm of decryption: $p = c \pmod{k}$, i.e. p - remainder from dividing c by k .

Remark 1 In our cryptosystem it is very easy to change the secret key. Indeed, if we change the key k by k_1 , after we get a new encryption cipher text value, but after decoding again we have $p = c_1 (\bmod k_1)$.

Definition 1 An integer s is called a partial key of pair (k, p) if it satisfies the conditions (1). If S is a finite set of different partial keys, then $|S|$ is called the period of S . If S is the infinite, the period is infinite.

Remark 2 a) Let k - secret key, p - the plain text. From theorem 1 it follows that for any partial key s , we can get the cipher text c_s . These cipher texts are different for different s , but $p = c_s (\bmod k)$ for every s .

b) The partial keys are used in the encryption process and not used in the decryption process.

It is obvious that for any finite set of pairs $(k, p_1), (k, p_2), ..., (k, p_m)$ there are an infinite number of intermediate keys.

Let $s_1, s_2, ..., s_m$ - set of different partial keys for pairs $(k, p_i), i = 1,2,...,m$. From the theorem 1 it follows that if we encrypt $p_1, p_2, ..., p_m$ by the algorithm (C_1) , then cipher texts $c_1, c_2, ..., c_m$ are different. We get the following:

(C_2) Algorithm of encryption: let k - secret key, $p_1, p_2, ..., p_m$ - set of plain texts, i.e. $|p_i| < |k|, i = 1,...,m$, $s_1, s_2, ..., s_m$ - set of different pairs of partial keys

$(k,p_1),(k,p_2),...,(k,p_m)$ respectively. We encrypt the plain text p_i, using the partial key s_i and an algorithm (C_1). In the result all cipher texts $c_1,c_2,...,c_m$ are different. Thus, we have: k - secret key, $p_1,p_2,...,p_m$ - plain text, $s_1,s_2,...,s_m$ - corresponding partial keys, $c_1,c_2,...,c_m$ - cipher texts.

(D_2) Algorithm of decryption:

$$p_i = c_i \ (\mathrm{mod}k),\ i = 1,2...,m.$$

Construction of partial keys

From the above it follows that the partial keys play an important role in the encryption algorithm. We propose two algorithms for constructing these keys.

I. Let $\Sigma = \{p_1,p_2,...,p_m, p_i \in Z\}$ - alphabet and $K = \max\{|p_i|, i = 1,2,...,m\}$. Then it is easy to see that

$$S = \{s \in Z, s \ \ is \ prime, (s,k) = 1, |s| > \max(K, \frac{k}{2})\}$$

Is a set of partial keys for all pairs $(k,p_i),\ i = 1,2,...,m$ and the period S is infinite.

II. Let be given the set of partial keys $S = \{s_1,s_2,...,s_l\}$ for pairs $(k,p_i),\ i = 1,2,...,m$. The next algorithm allows to extend S and get set of partial keys of any period $N \geq l$.

If we investigate the set $S_1 = \{s_1^{n_1}s_2^{n_2}...s_l^{n_l}, n_1 + n_2 + ... + n_l \geq 1\}$, it is obvious, that all of these products are partial keys for all pairs $(p,k_i),\ i = 1,2,...,m$. Therefore, if we add to S $(N-l)$ different elements of set S_1, then we get the set of partial keys $\overline{S}$ and period $\overline{S}$ is equal to N.

Attacks and modifications of algorithm

Consider the following attack on our cryptosystem.

$(*)$ Given a set of plain texts $p_1,p_2,...,p_n$, corresponding set of encrypted texts $c_1,c_2,...,c_n$ and secret key k. From algorithm of decryption implies the existence of integer numbers $d_1, ..., d_n$, such that

$$a_i = c_i - p_i = d_i\, k, \quad i = 1,2,...,n.$$

Then we have

$$\gcd(a_1, ..., a_n) = dk,\ d = \gcd(d_1, ..., d_n).$$

This attack can be effective for finding the secret key k.

The following modification of the encryption and decryption is proposed.

(C_3) Encryption algorithm: Let $a,b \in Z, |b| < |a\,k|$, where Z - the ring of integers and

$$\overline{c} = (q;r) \in Z \times Z$$

where $ac = bq + r$. Then $ac \neq r$, because $|c| > |k|$. Consider two maps $E_k : Z \to Z$ $\overline{E}_{a,b} : Z \to Z \times Z$, defined as follows:

$$E_k(p) = c,\ \overline{E}_{a,b}(c) = (q,r) \qquad (4)$$

Then

$$\overline{c} = \overline{E}_{a,b}(E_k(p)) \qquad (5)$$

We get: p - plain text, $\overline{c}$ - ciphertext. First step: we use the algorithm (C_2), second step: apply the mapping $\overline{E}_{a,b}$.

(D_3) Algorithm of decryption:

$$[a^{-1}(bq + r)](\mathrm{mod}k) = p \qquad (6)$$

Here we have three secret keys: k, a and b. Plain text – integer numbers cipher texts - the elements of the Cartesian product $Z \times Z$.

Obviously, if the cryptosystem with the encryption algorithm (C_2) and decryption algorithm (D_2) satisfies the properties (A), then the cryptosystem of algorithms (C_3) and (D_3) satisfies these properties.

Definition 2: A cryptosystem with encryption algorithm (C_3) and the decryption algorithm (D_3) is called EZ-cryptosystem.

Note 5: EZ-cryptosystem is a block cryptosystem

Suppose that p and $\overline{c} = (q;r)$ are known. Our goal is to find k, a, b, c or any information about them, provided with the known algorithms of encryption and decryption. We write: $k = x$, $a = y, b = z, c = t$, then we have following equations:

$$t(\mathrm{mod}x) = p \qquad (7)$$

$$yt = qz + r \qquad (8)$$

Well-known, that equation (7) with unknowns x, t has infinite set of solutions, but if x or t is fixed, then the equation (7) has unique solution. It is obvious, that the equation (8) has infinite set of solutions too.

Encryption application

Encryption application is used for encrypting the text and send it to the recipient. Thus, in the client-server automated system it works as a server application. The main window of the application is presented on Figure 1.

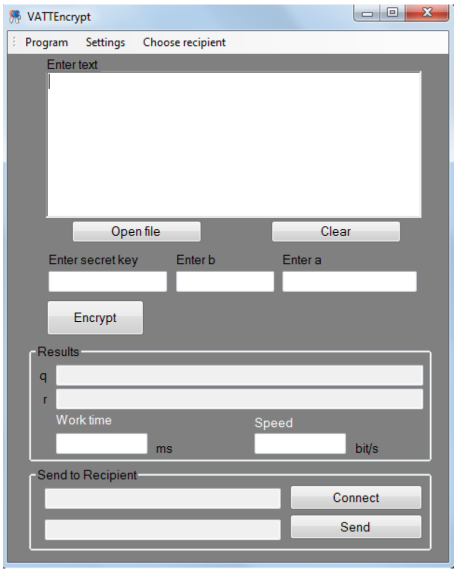


Figure 1 - Main window of encryption application

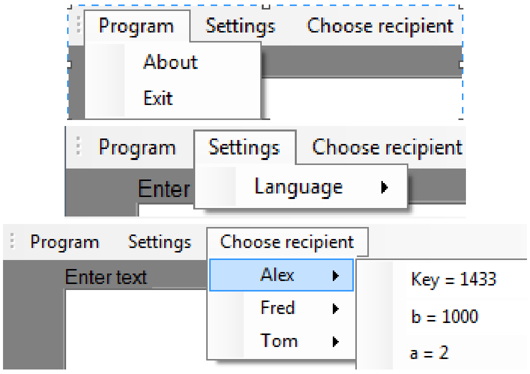
On the top of the window there is the menu panel (Figure 2)

The first inset "Program" used to watch the information about the author and program.

The inset "Settings" used for changing the language of the application. There are two languages available for the user: English and Russian.

The inset "Choose recipient" is used for changing the recipient of the message. There are three user created to show how the system works. For each user there are secret key k and additional keys a and b. They are written in the code of the program and applicable according to the main theorem of EZ-cryptosystem. While not any keys can be used, it could be difficult to find them. Therefore they are fixed before only for demonstration. Other applicable keys also could be used. In this case "Choose recipient" field should not be used.

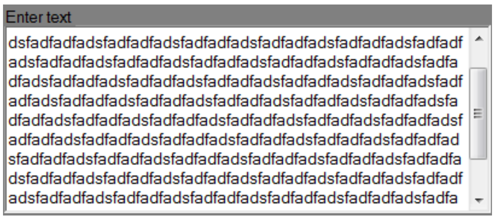
Menu panel is the same in the server and client applications.



Key = 1433
b = 1000
a = 2

Figure 2 - Menu panel

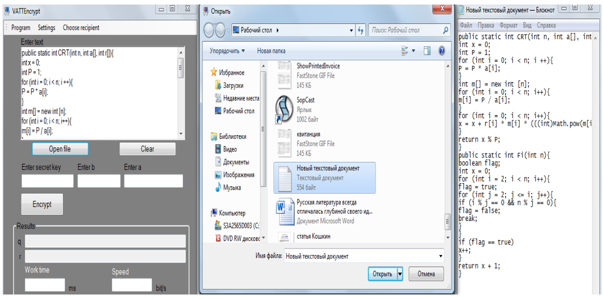
The field "Enter text" (Figure 3) used for writing the message. Message could have any size.



```
dsfadfadfadsfadfadfadsfadfadfadsfadfadfadsfadfadfadsfadfadf
adsfadfadfadsfadfadfadsfadfadfadsfadfadfadsfadfadfadsfadfa
dfadsfadfadfadsfadfadfadsfadfadfadsfadfadfadsfadfadfadsfadf
adfadsfadfadfadsfadfadfadsfadfadfadsfadfadfadsfadfadfadsfa
dfadfadsfadfadfadsfadfadfadsfadfadfadsfadfadfadsfadfadfadsf
adfadfadsfadfadfadsfadfadfadsfadfadfadsfadfadfadsfadfadfad
sfadfadfadsfadfadfadsfadfadfadsfadfadfadsfadfadfadsfadfadfa
dsfadfadfadsfadfadfadsfadfadfadsfadfadfadsfadfadfadsfadfadf
adsfadfadfadsfadfadfadsfadfadfadsfadfadfadsfadfadfadsfadfa
```

Figure 3 - "Enter text" field

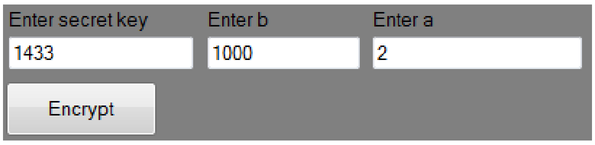
If the message is saved in the .txt file, it can be read from it. After choosing the needed file, its data will appear in the "Enter text" field (Figure 4).



```
public static int CRT(int n, int a[], int
int x = 0;
int P = 1;
for (int i = 0; i < n; i ++){
P = P * a[i];
}
int m[] = new int [n];
for (int i = 0; i < n; i++){
m[i] = P / a[i];
}
for (int i = 0; i < n; i++){
x = x + r[i] * m[i] * (((int)Math.pow(m[i
}
return x % P;
}
public static int Fi(int n){
boolean flag;
int x = 0;
for (int i = 2; i < n; i++){
flag = true;
for (int j = 2; j <= i; j++){
if (i % j == 0 && n % j == 0){
flag = false;
break;
}
}
if (flag == true)
x++;
}
return x + 1;
}
```

Figure 4 - Open file demonstration

Next, the keys should be chosen from "Choose recipient" inset or they could be entered by the keyboard without choosing the recipient. The encryption process could be finished by clicking the button "Encrypt" (Figure 5).



Enter secret key: 1433
Enter b: 1000
Enter a: 2

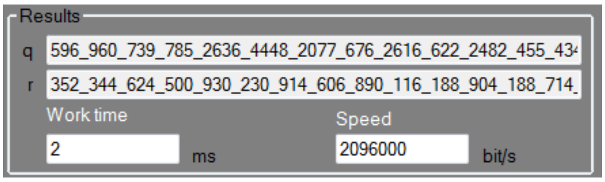
Figure 5 - Secret keys field and encryption

After the clicking the "Encrypt" button the message will be ciphered. The results panel(Figure 6) shows:

pair (q,r) values for each letter. In other words, it is encrypted text.

work time - is the time the program needed to encrypt the text.

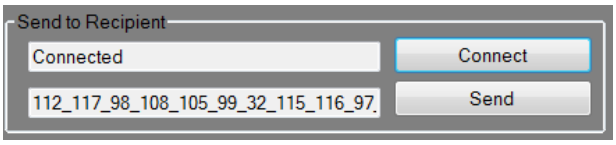
speed - is the number of bits that were encrypted in the interval of one second.



q 596_960_739_785_2636_4448_2077_676_2616_622_2482_455_43
r 352_344_624_500_930_230_914_606_890_116_188_904_188_714_
Work time 2 ms
Speed 2096000 bit/s

Figure 6 - Results panel

"Send to recipient" panel is used for the transmitting the message. After clicking the "Connect" button the application looks for the client, and, after it will be found, the text "Connected" will appear in the field near the button (Figure 7).

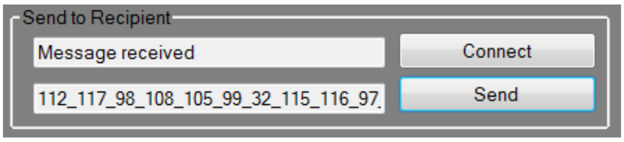


Connected

112_117_98_108_105_99_32_115_116_97

Figure 7 - Connection to the client

To send the message "Send" button is used. If the message is successfully sent, the text "Message received" will appear in the field to the left (Figure 8).



Message received

112_117_98_108_105_99_32_115_116_97

Figure 8 - Sending to the client

The last field shows the Unicode of the message. This field is not important for the user, while it shows the message prepared in the appropriate form for the encryption.

Decryption application is used for decrypting the text which is sent by the server. Thus, in the client-server automated system it works as a client application. The main window of the application is presented on figure 9.

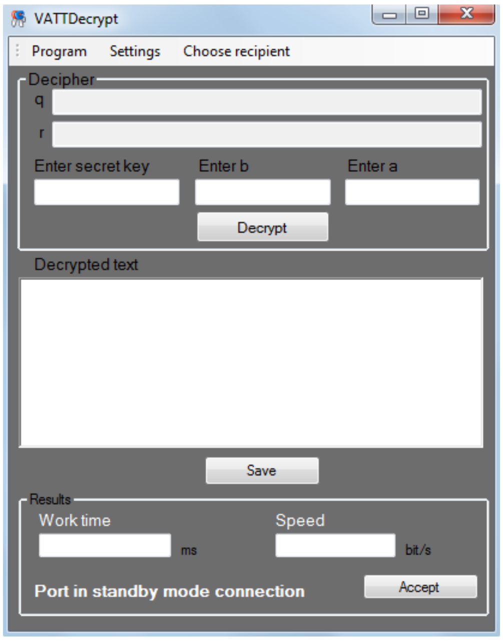
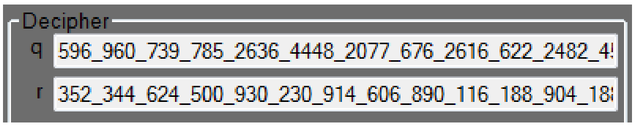


Figure 9 - Main window of decryption application

When the message is successfully sent to the client, the encrypted text values (q,r) will appear in fields (figure 10)

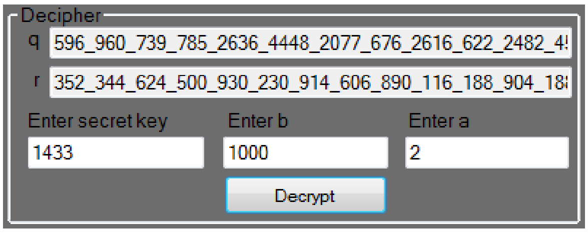


596_960_739_785_2636_4448_2077_676_2616_622_2482_4!

352_344_624_500_930_230_914_606_890_116_188_904_18‹

Figure 10 - Sent values panel

The secret keys could be chosen from "Choose recipient" panel or written manually. The recipient knows the sender of the message, therefore, the keys are also known. After choosing the key, "Decrypt" button is used for message decryption (Figure 11).



596_960_739_785_2636_4448_2077_676_2616_622_2482_4!

352_344_624_500_930_230_914_606_890_116_188_904_18‹

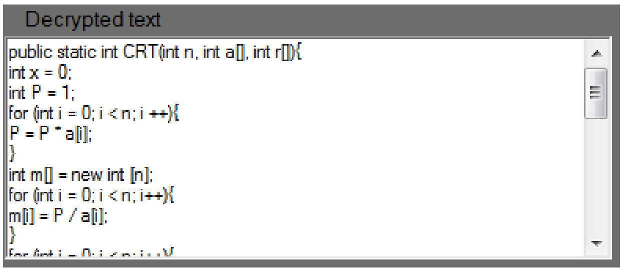
1433

1000

2

Figure 11 - Message decryption

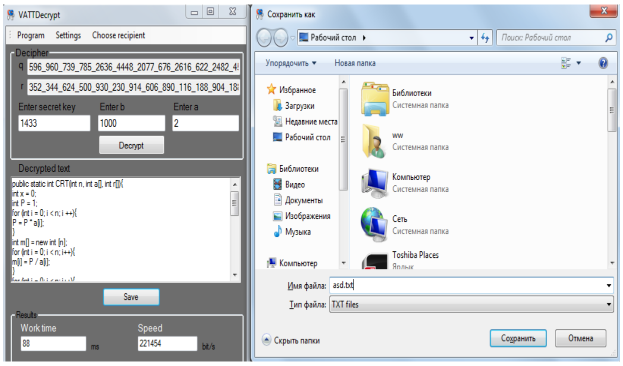
The decrypted text appears in "Decrypted text" field (Figure 12).



```
public static int CRT(int n, int a[], int r[]){
int x = 0;
int P = 1;
for (int i = 0; i < n; i ++){
P = P * a[i];
}
int m[] = new int [n];
for (int i = 0; i < n; i++){
m[i] = P / a[i];
}
```

Figure 12 - Decrypted text

The decrypted text could be copied manually from the textbox or the option "Save" could be used. The message could be saved to the file with .txt extension on the client computer (Figure 13).



596_960_739_785_2636_4448_2077_676_2616_622_2482_4!

352_344_624_500_930_230_914_606_890_116_188_904_18‹

Enter secret key: 1433

Enter b: 1000

Enter a: 2

```
public static int CRT(int n, int a[], int r[]){
int x = 0;
int P = 1;
for (int i = 0; i < n; i ++){
P = P * a[i];
}
int m[] = new int [n];
for (int i = 0; i < n; i++){
m[i] = P / a[i];
}
```

Work time: 88 ms

Speed: 221454 bit/s

Имя файла: asd.txt

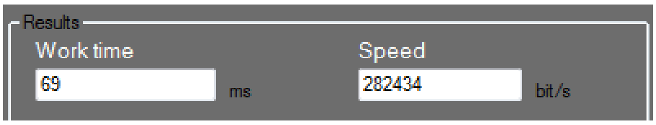
Тип файла: TXT files

Figure 13 - Message saving

The "Results" panel (Figure 14) shows:

work time - is the time the program needed to decrypt the text.

speed - is the number of bits that were decrypted in the interval of one second.



69

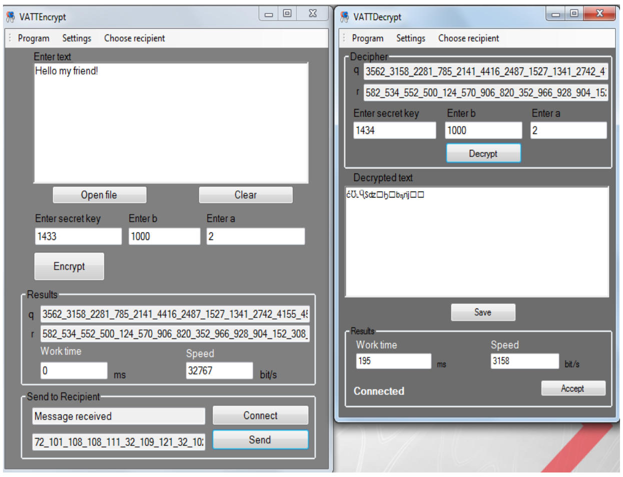
282434

Figure 14 - Results panel

"Accept" button used for finish the work of the decryption application.

Key importance

It is important for the receiver to know the message sender. In other case the wrong key could be entered and the message never be decrypted correctly (Figure 15).



Enter text
Hello my friend!

Enter secret key 1433
Enter b 1000
Enter a 2

Results
q 3562_3158_2281_785_2141_4416_2487_1527_1341_2742_4155_4
r 582_534_552_500_124_570_906_820_352_966_928_904_152_308_
Work time 0 ms
Speed 32767 bit/s

Send to Recipient
Message received
72_101_108_108_111_32_109_121_32_10

Decipher
q 3562_3158_2281_785_2141_4416_2487_1527_1341_2742_4
r 582_534_552_500_124_570_906_820_352_966_928_904_15
Enter secret key 1434
Enter b 1000
Enter a 2

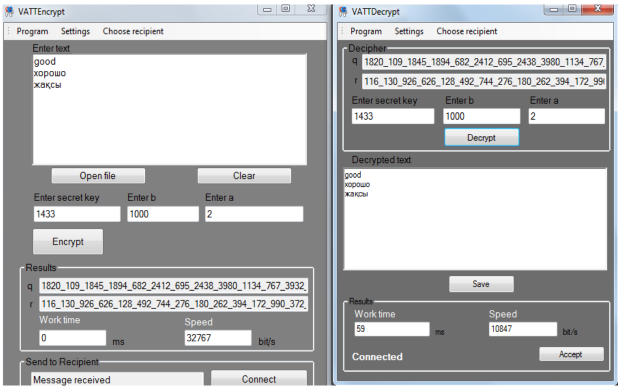
Decrypted text
ćՄ.ЧՏdz□ђ□bӊӈ□□

Results
Work time 195 ms
Speed 3158 bit/s
Connected

Figure 15 - Wrong key case

Available languages

The message in the program could be written on English, Russian or Kazakh language. The figure 16 shows the correct decryption of any of these three languages.



Enter text
good
хорошо
жақсы

Enter secret key 1433
Enter b 1000
Enter a 2

Results
q 1820_109_1845_1894_682_2412_695_2438_3980_1134_767_3932_
r 116_130_926_626_128_492_744_276_180_262_394_172_990_372_
Work time 0 ms
Speed 32767 bit/s

Send to Recipient
Message received

Decipher
q 1820_109_1845_1894_682_2412_695_2438_3980_1134_767_
r 116_130_926_626_128_492_744_276_180_262_394_172_99
Enter secret key 1433
Enter b 1000
Enter a 2

Decrypted text
good
хорошо
жақсы

Results
Work time 59 ms
Speed 10847 bit/s
Connected

Figure 16 - Available languages

These solutions make it possible to build a highly secure and reliable channels of personal data within the local network and the Internet (VPN), and solve a wide range of issues of information security, including the deployment of public key infrastructure (PKI) in corporate and government systems, distributed processing of personal data at any level of complexity.

The concept of building a virtual network VPN is quite a simple idea: if there are two Internet site to exchange information between two nodes it is necessary to construct virtual secure channel to ensure the confidentiality and integrity of information transmitted over public networks; access to the virtual channel should be extremely difficult all possible active and passive external observer.

CONCLUSION

The benefits received by the company from creating such virtual channels are primarily large financial savings, as in this case, the company may refuse to build or lease expensive dedicated communication channels to create its own intranet / extranet network and to use the cheap Internet channels, reliability and speed in which the majority is not inferior to the dedicated lines. The obvious cost-effectiveness of the introduction of MRI technology encourages enterprises to actively implement them.

Analysis of methods of protection of automated systems has allowed to formulate the requirements for a secure channel. During this analysis, we explored the VPN and ensure its security.Technology and software components of the project implemented by encryption and decryption algorithms coated EZ-cryptosystem.

References

[1] Dru Lavigne, VPN и IPSec на пальцах [Electronic resource]–2005.-http://www.nestor.minsk.by/sr/2005/03/050315.html - 07.06.2015

[2] A.Ten, V.Ten, Euclidean rings and cryptosystems Without repetition [Electronic resourse] – 2011. - http://www.tsi.lv/sites/default/files/editor/science/Research_journals/Computer/2011/V2/15_2_7_ten.pdf - 15.04.2015

[3] The Quantification of Operational Risk/ Paolo Vanini.// University of Zurich -2003. – 01.05.2015.

[4] Measuring operational risk in financial institutions: Contribution of credit risk modeling/ Georges Hübner/ - 2005. – 02.05.2015.

[5] Implications of Alternative Operational Risk Modeling Techniques/ Patrick de Fontnouvelle/ Eric Rosengren Federal Reserve Bank of Boston. -2004. – 04.05.2015.

[6] Cyclicality in catastrophic and operational risk measurements/ Linda Allen/ -2006. – 04.05.2015.

[7] Operational Risk: Measurement and Modeling/ Daniel J. Brown/ Professional Risk Managers International Association. – 2004. – 05.05.2015.